

? ss pn=DE 19734413
S3 1 SS PN=DE 19734413

? ts3/23/1

3/23/1 351 12240887 \$6.37 US

Derwent WPI

(c) 2004 Thomson Derwent. All rights reserved.

012240887 **Image available**

WPI Acc No: 1999-046994/199905

XRPX Acc No: N99-034283

**Upgrade process for database - ensures data compatibility
between old and new versions during upgrade and ensures data-bank tables
are updated under both versions during upgrade**

Patent Assignee: TELEFONAKTIEBOLAGET ERICSSON L M (TELF)

Number of Countries: 082 Number of Patents: 006

Abstract (Basic): DE 19734413 C

The process comprises the following steps: production of a new table in the database; copying of the data from the old database table into the new table; and addition of a new version of the previously stored procedure for processing data in the table.

Addition of an extra version of the procedure refers to both the previous and the upgrade procedure version. When data are to be processed, the extra procedure ensures it is processed in accordance with both the previous and the upgrade versions.

USE - Upgrading of database, while ensuring that downtime is minimised to minimum and that, should problem occur, upgrade can be rolled back and database returned to its original state.

ADVANTAGE - Other procedures require duplication of computer system hardware and software during upgrade. This is expensive and can cause licence problems. Invention uses extra process to ensure data added during upgrade process are transparent to both old and new database versions.

Dwg.3/4

Title Terms: UPGRADING; PROCESS; DATABASE; ENSURE; DATA; COMPATIBLE; NEW;
VERSION; UPGRADING; ENSURE; DATA; BANK; TABLE; UPDATE; VERSION; UPGRADING

Derwent Class: T01

International Patent Class (Main): G06F-017/30



19 BUNDESREPUBLIK
DEUTSCHLAND



DEUTSCHES
PATENT- UND
MARKENAMT

12 Patentschrift
10 DE 197 34 413 C 1

51 Int. Cl.⁶:
G 06 F 17/30

21 Aktenzeichen: 197 34 413.5-53
22 Anmeldetag: 8. 8. 97
43 Offenlegungstag: -
45 Veröffentlichungstag
der Patenterteilung: 7. 1. 99

Innerhalb von 3 Monaten nach Veröffentlichung der Erteilung kann Einspruch erhoben werden

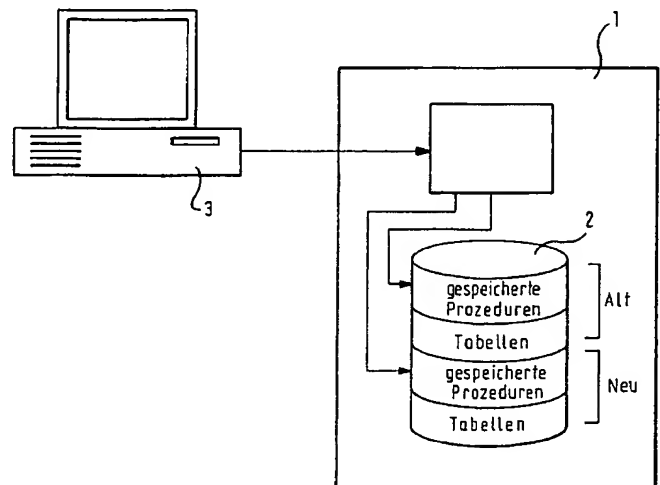
73 Patentinhaber:
Telefonaktiebolaget L M Ericsson, Stockholm, SE
74 Vertreter:
HOFFMANN · EITLE, 81925 München

72 Erfinder:
Sinander, Niklas, Västra Frölunda, SE

56 Für die Beurteilung der Patentfähigkeit in Betracht
gezogene Druckschriften:
US 54 40 735
EP 05 01 180 A2

54 Verfahren zum Aufrüsten einer Datenbank

57 Wenn eine Datenbank aufgerüstet wird (Upgrading), sollte sowohl die Stillstandszeit, als auch die Hardware- und Software-Anforderungen auf einem Minimum gehalten werden. Des weiteren sollte ein Umkehren (Rollback) des Upgrades jederzeit möglich sein. Das hier offenbarte Verfahren umfaßt die Schritte eines Erzeugens einer neuen Tabelle in der Datenbank, Kopieren und/oder Übertragen von Daten von der Tabelle in die neue Tabelle, Hinzufügen einer neuen Version der gespeicherten Prozedur zu der Datenbank, und Hinzufügen einer zusätzlich gespeicherten Prozedur zu der Datenbank, die auf sowohl die vorherige Version als auch die neue Version der gespeicherten Prozedur Bezug nimmt, wenn immer Daten mit der gespeicherten Prozedur verarbeitet werden sollen, um zu veranlassen, daß eine Datenverarbeitung in Übereinstimmung mit der vorherigen Version und der neuen Version der gespeicherten Prozedur stattfindet.



DE 197 34 413 C 1

DE 197 34 413 C 1

Die Erfindung betrifft Verfahren zum Aufrüsten einer Datenbank.

Relationale Datenbanken werden zum Handhaben des Speicherns und Wiedergewinnens von Daten für verschiedene Anwendungen und Umgebungen verwendet. Datenbanken dieser Art laufen auf Computersystemen. Wenn eine neue Funktionalität der Datenbank hinzugefügt werden soll, d. h. wenn die Datenbank aufrüstet wird, was auch als "Upgrading" bezeichnet wird, muß sehr oft auch das Format, in dem die Daten gespeichert sind, geändert werden. Üblicherweise muß während der Änderung ein Zugriff auf die Datenbank verwehrt werden, um Inkonsistenz zu vermeiden. Mit anderen Worten, es ist nicht erlaubt, Daten hinzuzufügen oder fortzuschreiben, während die Änderung der Funktionalität und des Formats der Datenbank stattfindet. Zumindestens ist die Datenbank während bestimmter Phasen eines Aufrüstens (Upgrade) nicht zugänglich, wobei die Phasen minimal sein sollten.

Ferner, wenn das Upgrade der Datenbank fehlschlägt, muß es möglich sein, zu dem vorherigen Zustand der Datenbank zurückzukehren. Datenänderungen, die während oder nach dem Upgrade durchgeführt worden sind, müssen in Betracht gezogen werden, sowie Datenänderungen, die dafür anstehen, durchgeführt zu werden.

Eine herkömmliche Lösung des zuvor erwähnten Problems ist die Verwendung eines zusätzlichen Systems während des Upgrades. Das zusätzliche System ist ein voll einsetzbares System mit all den Funktionalitäten des ursprünglichen Systems. Zusätzliche Hardware und Software wird benötigt, um alle Datenänderungen während des Upgrades der Datenbank einzutragen. Vor dem Upgrade wird die Datenbank auf das zusätzliche System kopiert, und ein Protokolliermechanismus (logging) wird zum Beispiel in dem ursprünglichen System initiiert. Das Upgrade der Datenbank wird dann in dem zusätzlichen System durchgeführt, während das ursprüngliche System damit weitermacht, wie zuvor zu arbeiten, aber auch alle Datenänderung während des Upgrades mitschreibt oder protokolliert. Wenn das Datenbank-Upgrade beendet ist, werden alle protokollierten Datenänderungen auf die aufrüstete Datenbank in dem zusätzlichen System angewendet. Wenn das Upgrade fehlschlägt, wird das zusätzliche System einfach getrennt, und das ursprüngliche System setzt seinen normalen Betrieb fort, jedoch ohne weiterhin zusätzlich Datenänderungen zu protokollieren.

Die obige Methode zum Aufrüsten einer Datenbank ist nachteilig, da ein zusätzliches System installiert und an die Stelle gebracht werden muß, wo die Datenbank aufrüstet werden soll. Während des Upgrades werden doppelte Softwarelizenzen benötigt, da auf beiden, dem ursprünglichen und dem zusätzlichen System, Datenbanken für eine bestimmte Zeit parallel laufen. Ferner stellt sich heraus, daß es recht kompliziert ist, das Datenbank-Upgrade aufgrund all der erforderlichen manuellen Eingriffe zu organisieren.

Aus US 5.440.735 ist ein Management-System für eine relationale Datenbank bekannt, das es dem Benutzer gestattet, auf einfache Weise Kopieroperationen festzulegen.

In EP 0 501 180 A2 wird das Problem des gleichzeitigen Verarbeitens von Transaktionen und Abfragen behandelt, bei dem sichergestellt sein muß, daß eine Abfrage nur auf solche Daten zugreift, die nicht durch eine gleichzeitig verarbeitete Transaktion während der Abfrage verändert werden. Zu jedem Datenobjekt wird eine begrenzte Zahl logischer Versionen dynamisch verwaltet, so daß die Abfragen auf einen aktuelleren Datenbestand zugreifen können, selbst wenn gleichzeitig Transaktionen den Datenbestand fortschreiben.

Deswegen ist das von der Erfindung zu lösende Problem, ein Verfahren zum Aufrüsten (Upgrading) einer Datenbank bereitzustellen, das eine Änderung des Datenbankformats beinhaltet, wobei das Verfahren zusätzliche Hardware und Software vermeidet, die Unterbrechung des Datenbankbetriebs auf ein Minimum beschränkt, und ein sicheres Zurückkehren in den ursprünglichen Zustand der Datenbank im Falle eines Fehlers während des Upgrades erlaubt.

Das obige Problem wird gelöst durch ein Verfahren zum Aufrüsten (Upgrading) einer Datenbank, die eine Tabelle zum Speichern von Daten und eine gespeicherte Prozedur zum Bearbeiten von Daten, die in der Tabelle der Datenbank gespeichert sind, verwendet, wobei die Datenbank auf einem Computersystem läuft, umfassend die Schritte:

- neues Erzeugen einer Tabelle in der Datenbank;
- Kopieren und/oder Übertragen von Daten von der Tabelle in die neue Tabelle;
- Hinzufügen einer neuen Version der gespeicherten Prozedur zu der Datenbank; und
- Hinzufügen einer zusätzlichen gespeicherten Prozedur zu der Datenbank, die sowohl auf die vorherige Version als auch die neue Version der gespeicherten Prozedur Bezug nimmt, wenn immer Daten mit der gespeicherten Prozedur bearbeitet werden sollen, um zu bewirken, daß eine Datenverarbeitung in Übereinstimmung mit der vorherigen Version und der neuen Version der gespeicherten Prozedur stattfindet.

Zum Herstellen einer Verbindung zu der zusätzlich gespeicherten Prozedur kann eine Systemtabelle verwendet werden. Als eine Alternative, oder zusätzlich, kann eine Konfigurationsdatei zum Erzeugen einer Verbindung zu der zusätzlich gespeicherten Prozedur verwendet werden. Die Konfigurationsdatei kann auf einem Speichermedium des Computersystems gespeichert werden.

Das obige Problem wird weiterhin gelöst durch ein Verfahren zum Aufrüsten (Upgrading) einer Datenbank durch Ersetzen einer alten Version einer Komponente der Datenbank, die Datenbankfunktionalität bereitstellt, durch eine neue Version, wobei die Datenbank auf einem Computersystem läuft, worin während des Aufrüstens die alte Version und die neue Version der Datenbankkomponente, die Datenbankfunktionalität bereitstellt, zumindest vorübergehend parallel verwendet werden, und wobei Bezug genommen wird auf beide Versionen über eine zusätzliche Komponente der Datenbank, die eine Verbindung zu beiden Versionen der Datenbankkomponente bereitstellt.

Die Verbindung kann mit Hilfe einer Systemtabelle bereitgestellt werden, die einen Zugriff auf die Komponente der Datenbank auf die zusätzliche Komponente weiterleitet. Die Systemtabelle kann eine Tabelle in der Datenbank sein.

Zusätzlich, oder als eine Alternative, kann die Verbindung auch mit Hilfe einer Konfigurationsdatei bereitgestellt werden, die einen Zugriff auf die Komponente der Datenbank auf die zusätzliche Komponente weiterleitet. Die Konfigurationsdatei wird vorteilhafterweise auf einem Speichermedium des Computersystems gespeichert.

Das obige Problem wird schließlich gelöst durch ein Verfahren zur Aufrüstung einer Datenbank, die eine Tabelle zum

Speichern von Daten verwendet, wobei die Datenbank auf einem Computersystem läuft, umfassend die Schritte:

- Erzeugen einer neuen Tabelle in der Datenbank;
- Definieren einer Datenmenge einer vorbestimmten Granularität innerhalb der Tabelle, basierend auf der zum Kopieren und/oder Übertragen von Daten in die neue Tabelle benötigten Zeit, während der nicht auf die Datenmenge zugegriffen werden kann; und
- Kopieren und/oder Übertragen von Daten von der Tabelle in die neue Tabelle, wobei die Datenmengen gemeinsam gehandhabt werden.

Die Komponenten der Datenbank sind gespeicherte Prozeduren, Trigger, Indizes oder Tabellen. Die neue Version der Tabelle kann ein von dem Format der alten Version der Tabelle verschiedenes Format haben.

Die Erfindung wird in genaueren Einzelheiten weiter unten beschrieben, wo auf die Abbildungen Bezug genommen wird.

Fig. 1 zeigt ein Diagramm eines Computersystems, das eine Datenbank und Arbeitsplatzrechner, die auf die Datenbank zugreifen, umfaßt.

Fig. 2a zeigt ein Diagramm, das ein Aufrüsten einer Datenbank auf Tabellenebene gemäß einer Ausbildung der Erfindung darstellt.

Fig. 2b zeigt ein Diagramm, das ein Aufrüsten einer Datenbank auf Seitenebene gemäß einer Ausbildung der Erfindung darstellt.

Fig. 3 zeigt ein Diagramm eines Computersystems, das eine Systemtabelle zum Aufrüsten einer Datenbank gemäß einer Ausbildung der Erfindung verwendet.

Fig. 4 zeigt ein Diagramm des Computersystems, das eine Konfigurationsdatei zum Aufrüsten einer Datenbank gemäß einer Ausbildung der Erfindung verwendet.

Fig. 1 zeigt ein Computersystem 1, das eine Datenbank 2 auf einem geeigneten Speichermedium, beispielsweise einer Festplatte oder einer Gruppe von Festplatten, umfaßt. Benutzer greifen auf die Information oder auf die in der Datenbank gespeicherten Daten zu, zum Beispiel von einer Eingabestation oder von deren Arbeitsplatzrechnern 3a, 3b, . . ., die mit dem Computersystem 1 entweder direkt oder über ein lokales Netzwerk oder Weitverkehrsnetzwerk (LAN oder WAN) verbunden sind. Die Datenbank 2 umfaßt Tabellen, gespeicherte Prozeduren, Trigger, Indices, usw., die jeweils alle gewisse Datenbankfunktionen bereitstellen.

Wenn ein Upgrade der Datenbank benötigt wird, werden Änderungen für alle oder für einige der Komponenten, die Datenbankfunktionalität bereitstellen, durchgeführt. Gewöhnlicherweise werden neue und/oder zusätzliche Funktionen mit Hilfe von neuen und/oder zusätzlich gespeicherten Prozeduren, Triggern usw. bereitgestellt. Dies erfordert manchmal auch, daß Änderungen des Tabellenformats der Datenbank durchgeführt werden.

Gemäß der Erfindung werden neue Tabellen erzeugt, um die Daten aufzunehmen, die in den alten Tabellen gespeichert sind, die die Tabellen sind, die von der Datenbank bis zum Upgrade der Datenbank benutzt werden. Daten werden in die neuen Tabellen kopiert und/oder übertragen, wobei während dieses Vorgangs die Daten, falls erforderlich, an das neue Format der Tabellen angepaßt werden. Um Stillstandszeit zu vermeiden, d. h. Perioden in denen der Benutzer nicht auf die Datenbank zugreifen kann, benötigt der Prozeß des Kopierens und/oder Übertragens von Daten von den alten Tabellen auf die neuen Tabellen besondere Beachtung. Gemäß der Erfindung werden Datenmengen einer vorbestimmten Granularität innerhalb der Tabellen definiert, wobei die Datenmengen gemeinsam während dem Upgrade-Prozeß behandelt werden, um die Zeitspannen, in denen das System nicht zugänglich ist, zu reduzieren.

Wie in Fig. 2a gezeigt, wenn ein Upgrade auf Tabellenebene (Granularität I) basiert, wird die gesamte aufzurüstende Tabelle in einem Arbeitsgang modifiziert. Dieser Lösungsweg sollte gewählt werden, wenn die Modifikation ein schneller Vorgang ist, zum Beispiel des Hinzufügens eines vorgegebenen Wertes zu einer Spalte. Während eine Tabelle aufgerüstet wird, ist der Zugriff auf die Tabelle als Ganzes verwehrt, um Inkonsistenzen zu vermeiden. Wie in Fig. 2b gezeigt, wenn die Modifikation einer Tabelle ein komplizierterer Vorgang ist, der lange dauert, wird die Tabelle auf Seitenebene (Granularität II) aufgerüstet. Eine Seite (page) kann verstanden werden als eine interne Systemeinheit einer speziellen Größe, beispielsweise 1024 Byte, die gewöhnlicherweise – vollständig oder teilweise – mindestens eine oder mehrere in Beziehung stehende Informationseinheiten, wie beispielsweise Felder oder Datensätze umfaßt. Durch Verwehren eines Zugriffs auf die Seite wird mindestens ein Datensatz der Tabelle gesperrt. Zum Aufrüsten auf Seitenebene wird eine neue Tabelle installiert, und Daten von der alten Tabelle werden in die neue Tabelle, wie durch den Operator T in Fig. 2b gezeigt, kopiert und/oder übertragen. Während die Tabelle aufgerüstet wird, wird ein Zugriff auf die Seite, die momentan von dem Operator T bearbeitet wird, verwehrt, um Inkonsistenz zu vermeiden.

Des weiteren müssen sowohl die alten als auch die neuen Tabellen synchronisiert werden, soweit Datenänderungen und neue Einträge betroffen sind. Durch Aktualisieren sowohl der neuen als auch der alten Tabellen, wie von der Erfindung vorgeschlagen, wird nicht nur Datenkonsistenz erreicht, sondern auch die Möglichkeit geschaffen, das Upgrade aufzugeben und den Betrieb des ursprünglichen Systems fortzusetzen. Das Upgrade der Daten in der alten und neuen Tabelle ist ein atomarer Vorgang, d. h. Zugriff auf die Tabelle oder die Seite ist verwehrt, bis beide Tabellen aufgerüstet sind. Vorzugsweise wird ein Datenupgrade in den neuen Tabellen nur dort durchgeführt, wo Datensätze bereits von den alten Tabellen in die neuen Tabellen kopiert und/oder übertragen worden sind.

Um ein Aufrüsten von Tabellen einer Datenbank gemäß der Erfindung durchzuführen, müssen die verbleibenden Komponenten, die Datenbankfunktionalitäten wie gespeicherte Prozeduren, Trigger, Indizes, usw. bereitstellen, sowohl in einer alten Version als auch in einer neuen Version existieren. Um jeglichen Konflikt während des Zugriffs auf eine Komponente der Datenbank, wie gespeicherte Prozeduren, zu verhindern, wird ein Verfahren zum Handhaben verschiedener Versionen von gespeicherten Prozeduren oder dergleichen benötigt. Drei Verfahren werden im folgenden diskutiert.

Ein erstes Verfahren um Versionen von Datenbankkomponenten handzuhaben, wird mit besonderer Betonung auf gespeicherten Prozeduren der Datenbank beschrieben. Der Fachmann wird verstehen, daß dieser Prozeß auch für andere

Komponenten einer Datenbank, vergleichbar den gespeicherten Prozeduren, durchgeführt werden kann.

Wie in Fig. 3 gezeigt, wird eine Systemtabelle während dem Upgrade-Prozeß verwendet, wobei die Systemtabelle Bezugnahmen enthält auf alle oder nur diejenigen gespeicherten Prozeduren, die aufgerüstet werden sollen, d. h. durch gespeicherte Prozeduren einer neuen Version ersetzt werden sollen. Auf alle oder nur diese gespeicherten Prozeduren, die aufgerüstet werden sollen, wird über die Systemtabelle während des Upgrade-Prozesses verwiesen. Eine Systemtabelle für das Verfahren gemäß der Erfindung kann folgendermaßen aussehen:

Tabelle 1

Dem System bekannter Name der gespeicherten Prozedur	Basisversion	Zielversion	Upgrade- Version
sp_a	sp_a_1.0	sp_a_1.1	sp_a_upgr
sp_b	sp_b	-	-
sp_c	sp_c_1.0	sp_c_1.1	sp_c_upgr
...

Im Normalbetrieb verweisen die Eintragungen in der Systemtabelle nur auf einen Satz gespeicherter Prozeduren, d. h. auf die Basisversion. Wenn ein Arbeitsplatzrechner auf die Datenbank zugreift, schaut das System den momentan gültigen Namen oder Version der gespeicherten Prozedur nach, mit dem die durch die Workstation ausgelöste spezifische Transaktion verbunden ist. Wenn zum Beispiel der Zugriff von einer Workstation erfordert, daß gespeicherte Prozedur sp_a abgearbeitet wird, wird das System die gültige Version der gespeicherten Prozedur sp_a in der Systemtabelle nachschauen. In der obigen Tabelle, und im Normalbetrieb ist sp_a_1.0 die gültige Version der gespeicherten Prozedur sp_a, die dann von dem System verwendet wird, um die durch den Arbeitsplatzrechner verlangte Transaktion durchzuführen.

Im Falle eines Upgrades der Datenbank müssen neue Versionen aller oder einiger gespeicherter Prozeduren in der Datenbank installiert werden. Es ist auch möglich, daß eine neue gespeicherte Prozedur der Datenbank hinzugefügt wird. Gemäß der Erfindung wird die Systemtabelle mit Namen/Versionen der neuen gespeicherten Prozedur aufgerüstet, d. h. die Zielversionen werden hinzugefügt. Um alte und neue Tabellen der Datenbank synchron zu halten, werden zusätzliche gespeicherte Prozeduren sp_a_upgr und sp_c_upgr in die Systemtabelle hinzugefügt. Die zusätzlichen gespeicherten Prozeduren sp_a_upgr und sp_c_upgr werden nur während des Upgrade-Prozesses verwendet, und nutzen die neuen und die alten gespeicherten Prozeduren, um die neuen und die alten Tabellen in Übereinstimmung mit den durch die Arbeitsplatzrechner ausgelösten Transaktionen zu aktualisieren.

Während eines Upgrades initiiert beispielsweise die zusätzlich gespeicherte Prozedur sp_a_upgr in der Datenbank zwei Bearbeitungszweige, basierend auf der Basisversion sp_a_1.0 und der neuen Version sp_a_1.1, wodurch Daten in den neuen und den alten Tabellen des Datenbanksystems bearbeitet und fortgeschrieben werden. Bezüglich der gespeicherten Prozedur sp_b gibt es keine zusätzliche gespeicherte Prozedur, da die gespeicherte Prozedur sp_b nicht aufgerüstet wird, d. h. in der alten und der neuen Datenbank gleich bleibt. Das System wird in solchen Fällen auf die Basisversion sp_b verweisen, und die Basisversion für die alten und die neuen Tabellen verwenden.

Der Wechsel von den alten gespeicherten Prozeduren auf die neuen gespeicherten Prozeduren besteht darin den Zugriff auf die Systemtabelle zu verwehren, die Verweise aufzurufen, so daß auf die gewünschten gespeicherten Prozeduren verwiesen wird, um den Zugriff auf die Systemtabelle zu gewähren. Das Aufrüsten von Verweisen ist ein sehr schneller Vorgang, so daß Benutzer des Systems nur eine sehr kleine Verzögerung und keine Stillstandzeit bemerken.

Ein zweites Verfahren um Versionen von Datenbankkomponenten handzuhaben wird im folgenden beschrieben. Wie in Fig. 4 gezeigt, wird eine Konfigurationsdatei während des Upgrade-Prozesses verwendet, wobei die Datei auf alle oder nur jene gespeicherten Prozeduren verweist, die aufgerüstet werden sollen, d. h. durch gespeicherte Prozeduren einer neuen Version ersetzt werden. Auf alle oder nur jene gespeicherte Prozeduren, die aufgerüstet werden sollen, wird über die Konfigurationsdatei während des Upgrade-Prozesses Bezug genommen. Die Konfigurationsdatei wird in einem separaten Speichermedium 4, vorgesehen in dem Computersystem 1, oder in dem Speichermedium, auf dem auch die Datenbank 2 gespeichert ist, gespeichert.

Bevor auf die Datenbank durch eine Entität (z. B. eine Anwendung) von einem Arbeitsplatzrechner 3 zugegriffen wird, wird die Konfigurationsdatei gelesen, um zu sehen, welche gespeicherte Prozedur zu verwenden ist. Wenn es keinen Eintrag in der Konfigurationsdatei für eine spezifische gespeicherte Prozedur gibt, wird die dem System bekannte gespeicherte Prozedur verwendet. Eine Konfigurationsdatei gemäß der Erfindung kann eine einfache Textdatei sein, und kann wie folgt aussehen:

Tabelle 2

#UPGRADE-KONFIGURATIONSDATEI

#

#EINTRÄGE: dem System bekannter Name der gespeicherten

#Prozedur; Basisversion; Zielversion; Upgrade-Version

sp_a; sp_a_1.0; sp_a_1.1; sp_a_upgr

sp_b; sp_b

sp_c; sp_c_1.0; sp_c_1.1; sp_c_upgr

...

Im Normalbetrieb verweisen die Einträge in der Konfigurationsdatei nur auf einen Satz von gespeicherten Prozeduren, nämlich auf die Basisversion. Wenn ein Arbeitsplatzrechner auf die Datenbank zugreift, schaut das System den momentan gültigen Namen oder Version der gespeicherten Prozedur nach, mit der die durch den Arbeitsplatzrechner ausgelöste spezifische Transaktion verbunden ist. Wenn der Zugriff von einem Arbeitsplatzrechner beispielsweise erfordert, daß die gespeicherte Prozedur sp_a abgearbeitet wird, wird das System die gültige Version der gespeicherten Prozedur sp_a in der Konfigurationsdatei nachschauen. In der obigen Datei ist sp_a_1.0 die gültige Version der gespeicherten Prozedur sp_a, die dann von dem System verwendet wird, um die durch den Arbeitsplatzrechner verlangte Transaktion auszuführen.

Im Falle eines Upgrades der Datenbank müssen neue Versionen von allen oder einigen gespeicherten Prozeduren in der Datenbank installiert werden. Es ist auch möglich, daß eine neue gespeicherte Prozedur der Datenbank hinzugefügt wird. Um alte und neue Tabellen der Datenbank synchron zu halten, werden zusätzliche gespeicherte Prozeduren sp_a_upgr und sp_c_upgr der Konfigurationsdatei hinzugefügt. Die zusätzlichen gespeicherten Prozeduren sp_a_upgr und sp_c_upgr werden nur während des Upgrade-Prozesses verwendet, und nutzen die neuen und alten gespeicherten Prozeduren, um die neuen und alten Tabellen in Übereinstimmung mit den durch den Arbeitsplatzrechner ausgelösten Transaktionen auszuführen.

Während eines Upgrades löst beispielsweise die zusätzliche gespeicherte Prozedur sp_a_upgr an dem Arbeitsplatzrechner zwei Bearbeitungszweige aus, basierend auf der Basisversion sp_a_1.0 und der neuen Version sp_a_1.1, wodurch Daten bearbeitet in den neuen und den alten Tabellen des Datenbanksystems bearbeitet und fortgeschrieben werden. Bezugnehmend auf die gespeicherte Prozedur sp_b existiert keine zusätzliche gespeicherte Prozedur, da gespeicherte Prozedur sp_b nicht aufgerüstet wird, d. h. in der alten und der neuen Datenbank gleich bleibt. Das System wird in solchen Fällen auf die Basisversion sp_b Bezug nehmen, und die Basisversion für die neuen und die alten Tabellen verwenden.

Der Wechsel von den alten gespeicherten Prozeduren zu den neuen gespeicherten Prozeduren wird durch Aufrüsten der Konfigurationsdatei und erneutes Lesen durchgeführt. Ein Aufrüsten und erneutes Lesen der Konfigurationsdatei ist ein sehr schneller Vorgang, so daß die Benutzer des Systems nur eine sehr kleine Verzögerung und eine kleine Stillstandzeit bemerken werden.

Ein drittes Verfahren um Versionen von Datenbankkomponenten handzuhaben könnte eine Kombination aus einer Systemtabelle und einer Konfigurationsdatei sein. Die Systemtabelle wird auf einige gespeicherte Prozeduren verweisen, während die Konfigurationsdatei nur auf andere gespeicherte Prozeduren verweisen wird.

Die anderen Komponenten der Datenbank können auf gleiche Weise gehandhabt werden. Um beispielsweise in der Lage zu sein, zwei oder mehrere Versionen der Tabellen zu halten, kann auf die Tabellen über die Systemtabelle oder die Konfigurationsdatei verwiesen werden, die in ähnlicher Weise behandelt werden wie zuvor im Hinblick auf gespeicherte Prozeduren erläutert.

Patentansprüche

1. Verfahren zum Aufrüsten einer Datenbank, die eine Tabelle zum Speichern von Daten und eine gespeicherte Prozedur zum Bearbeiten von Daten, die in der Tabelle der Datenbank gespeichert werden, verwendet, wobei die Datenbank auf einem Computersystem läuft, umfassend die Schritte:

- neues Erzeugen einer Tabelle in der Datenbank;
- Kopieren und/oder Übertragen von Daten von der Tabelle in die neue Tabelle;
- Hinzufügen einer neuen Version (sp_a_1.1, sp_b_1.1) der gespeicherten Prozedur zu der Datenbank; und
- Hinzufügen einer zusätzlichen gespeicherten Prozedur (sp_a_upgr, sp_b_upgr, . . .) zu der Datenbank, die sowohl auf die vorherige Version (sp_a_1.0, sp_b_1.0) als auch die neue Version (sp_a_1.1, sp_b_1.1) der gespeicherten Prozedur Bezug nimmt, wenn immer Daten mit der gespeicherten Prozedur bearbeitet werden sollen, um zu bewirken, daß eine Datenverarbeitung in Übereinstimmung mit der vorherigen Version (sp_a_1.0, sp_b_1.0) und der neuen Version (sp_a_1.1, sp_b_1.1) der gespeicherten Prozedur stattfindet.

2. Verfahren zum Aufrüsten einer Datenbank nach Anspruch 1, dadurch gekennzeichnet, daß eine Systemtabelle eine Verbindung zu der zusätzlichen gespeicherten Prozedur (sp_a_upgr, sp_c_upgr) erzeugt.

3. Verfahren zum Aufrüsten einer Datenbank nach Anspruch 2, dadurch gekennzeichnet, daß die Systemtabelle eine Tabelle der Datenbank ist.

4. Verfahren zum Aufrüsten einer Datenbank nach Anspruch 1, 2 oder 3, dadurch gekennzeichnet, daß eine Konfigurationsdatei eine Verbindung zu der zusätzlich gespeicherten Prozedur (sp_a_upgr, sp_c_upgr) herstellt.
5. Verfahren zum Aufrüsten einer Datenbank nach Anspruch 4, dadurch gekennzeichnet, daß die Konfigurationsdatei auf einem Speichermedium (4) des Computersystems gespeichert wird.
6. Verfahren zum Aufrüsten einer Datenbank durch Ersetzen einer alten Version einer Datenbankkomponente, die Datenbankfunktionalität bereitstellt, durch eine neue Version, wobei die Datenbank auf einem Computersystem läuft, dadurch gekennzeichnet, daß während des Aufrüstens die alte Version (sp_a_1.0, sp_c_1.0) und die neue Version (sp_a_1.1, sp_c_1.1) der Datenbankkomponente, die Datenbankfunktionalität bereitstellt, zumindest zeitweise parallel verwendet werden, und daß auf beide Versionen über eine zusätzliche Komponente (sp_a_upgr, sp_b_upgr) der Datenbank, die eine Verbindung zu beiden Versionen der Datenbankkomponente bereitstellt, Bezug genommen wird.
7. Verfahren nach Anspruch 6, dadurch gekennzeichnet, daß die Verbindung mit Hilfe einer Systemtabelle bereitgestellt wird, die einen Zugriff auf die Komponente (sp_a, sp_c) der Datenbank auf die zusätzliche Komponente (sp_a_upgr, sp_c_upgr) weiterleitet.
8. Verfahren nach Anspruch 7, dadurch gekennzeichnet, daß die Systemtabelle eine Tabelle der Datenbank ist.
9. Verfahren nach Anspruch 6, 7 oder 8, dadurch gekennzeichnet, daß die Verbindung mit Hilfe einer Konfigurationsdatei bereitgestellt wird, die einen Zugriff auf die Komponente (sp_a, sp_c) der Datenbank auf die zusätzliche Komponente (sp_a_upgr, sp_c_upgr) weiterleitet.
10. Verfahren nach Anspruch 9, dadurch gekennzeichnet, daß die Konfigurationsdatei auf einem Speichermedium (4) des Computersystems gespeichert ist.
11. Verfahren nach einem der Ansprüche 6 bis 10, dadurch gekennzeichnet, daß die Komponente der Datenbank eine gespeicherte Prozedur ist.
12. Verfahren nach einem der Ansprüche 6 bis 10, dadurch gekennzeichnet, daß die Komponente der Datenbank ein Trigger ist.
13. Verfahren nach einem der Ansprüche 6 bis 10, dadurch gekennzeichnet, daß die Komponente der Datenbank ein Index ist.
14. Verfahren nach einem der Ansprüche 6 bis 10, dadurch gekennzeichnet, daß die Komponente der Datenbank eine Tabelle ist.
15. Verfahren nach Anspruch 14, dadurch gekennzeichnet, daß die neue Version der Tabelle ein Format hat, das von dem Format der alten Version der Tabelle verschieden ist.
16. Verfahren zum Aufrüsten einer Datenbank, die eine Tabelle zum Speichern von Daten verwendet, wobei die Datenbank auf einem Computersystem läuft, umfassend die Schritte:
 - Erzeugen einer neuen Tabelle in der Datenbank;
 - Definieren einer Datenmenge einer vorbestimmten Granularität innerhalb der Tabelle, basierend auf der zum Kopieren und/oder Übertragen von Daten in die neue Tabelle benötigten Zeit, während der nicht auf die Datenmenge zugegriffen werden kann; und
 - Kopieren und/oder Übertragen von Daten von der Tabelle in die neue Tabelle, wobei die Datenmengen gemeinsam gehandhabt werden.
17. Verfahren nach Anspruch 16, dadurch gekennzeichnet, daß die Datenmenge die Tabelle ist.
18. Verfahren nach Anspruch 16, dadurch gekennzeichnet, daß die Datenmenge eine Seite der Tabelle ist.
19. Verfahren nach einem der Ansprüche 16 bis 18, dadurch gekennzeichnet, daß die Datenbank eine gespeicherte Prozedur zum Bearbeiten von Daten, die in der Tabelle der Datenbank gespeichert sind, verwendet, umfassend die Schritte:
 - Hinzufügen einer neuen Version (sp_a_1.1, sp_b_1.1) der gespeicherten Prozedur zu der Datenbank;
 - Hinzufügen einer zusätzlichen gespeicherten Prozedur (sp_a_upgr, sp_b_upgr, ...) zu der Datenbank, die auf sowohl die vorherige Version (sp_a_1.0, sp_b_1.0) als auch die neue Version (sp_a_1.1, sp_b_1.1) der gespeicherten Prozedur Bezug nimmt, wenn immer Daten mit der gespeicherten Prozedur verarbeitet werden sollen, um zu bewirken, daß eine Datenverarbeitung in Übereinstimmung mit der vorherigen Version (sp_a_1.0, sp_b_1.0) und der neuen Version (sp_a_1.1, sp_b_1.1) der gespeicherten Prozedur stattfindet.
20. Verfahren zum Aufrüsten einer Datenbank nach Anspruch 19, dadurch gekennzeichnet, daß eine Systemtabelle eine Verbindung zu der zusätzlichen gespeicherten Prozedur (sp_a_upgr, sp_c_upgr) herstellt.
21. Verfahren zum Aufrüsten einer Datenbank nach Anspruch 20, dadurch gekennzeichnet, daß die Systemtabelle eine Tabelle der Datenbank ist.
22. Verfahren zum Aufrüsten einer Datenbank nach Anspruch 19, 20 oder 21, dadurch gekennzeichnet, daß eine Konfigurationsdatei eine Verbindung zu der zusätzlichen gespeicherten Prozedur (sp_a_upgr, sp_c_upgr) herstellt.
23. Verfahren zum Aufrüsten einer Datenbank nach Anspruch 22, dadurch gekennzeichnet, daß die Konfigurationsdatei auf einem Speichermedium (4) des Computersystems gespeichert ist.
24. Verfahren zum Aufrüsten einer Datenbank nach den Ansprüchen 16 bis 18, umfassend die Schritte des Ersetzens einer alten Version einer Datenbankkomponente, die Datenbankfunktionalität bereitstellt, durch eine neue Version, wobei während des Upgrades die alte Version (sp_a_1.0, sp_c_1.0) und die neue Version (sp_a_1.1, sp_c_1.1) der Datenbankkomponente, die Datenbankfunktionalität bereitstellt, zumindest vorübergehend parallel verwendet werden, und wobei auf beide Versionen über eine zusätzliche Komponente (sp_a_upgr, sp_b_upgr) der Datenbank, die eine Verbindung mit beiden Versionen der Datenbankkomponente bereitstellt, Bezug genommen wird.
25. Verfahren nach Anspruch 24, dadurch gekennzeichnet, daß die Verbindung mit Hilfe einer Systemtabelle bereitgestellt wird, die einen Zugriff auf die Komponente (sp_a, sp_c) der Datenbank auf die zusätzliche Komponente (sp_a_upgr, sp_c_upgr) weiterleitet.
26. Verfahren nach Anspruch 25, dadurch gekennzeichnet, daß die Systemtabelle eine Tabelle der Datenbank ist.
27. Verfahren nach Anspruch 24, 25 oder 26, dadurch gekennzeichnet, daß die Verbindung mit Hilfe einer Konfi-

DE 197 34 413 C 1

gurationsdatei bereitgestellt wird, die einen Zugriff auf die Komponente (sp_a, sp_c) der Datenbank auf die zusätzliche Komponente (sp_a_upgr, sp_c_upgr) weiterleitet.

28. Verfahren nach Anspruch 27, dadurch gekennzeichnet, daß die Konfigurationsdatei auf einem Speichermedium (4) des Computersystems gespeichert wird.

29. Verfahren nach einem der Ansprüche 24 bis 28, dadurch gekennzeichnet, daß die Komponente der Datenbank eine gespeicherte Prozedur ist. 5

30. Verfahren nach einem der Ansprüche 24 bis 28, dadurch gekennzeichnet, daß die Komponente der Datenbank ein Trigger ist.

31. Verfahren nach einem der Ansprüche 24 bis 28, dadurch gekennzeichnet, daß die Komponente der Datenbank ein Index ist. 10

32. Verfahren nach einem der Ansprüche 24 bis 28, dadurch gekennzeichnet, daß die Komponente der Datenbank eine Tabelle ist.

Hierzu 4 Seite(n) Zeichnungen

15

20

25

30

35

40

45

50

55

60

65

- Leerseite -

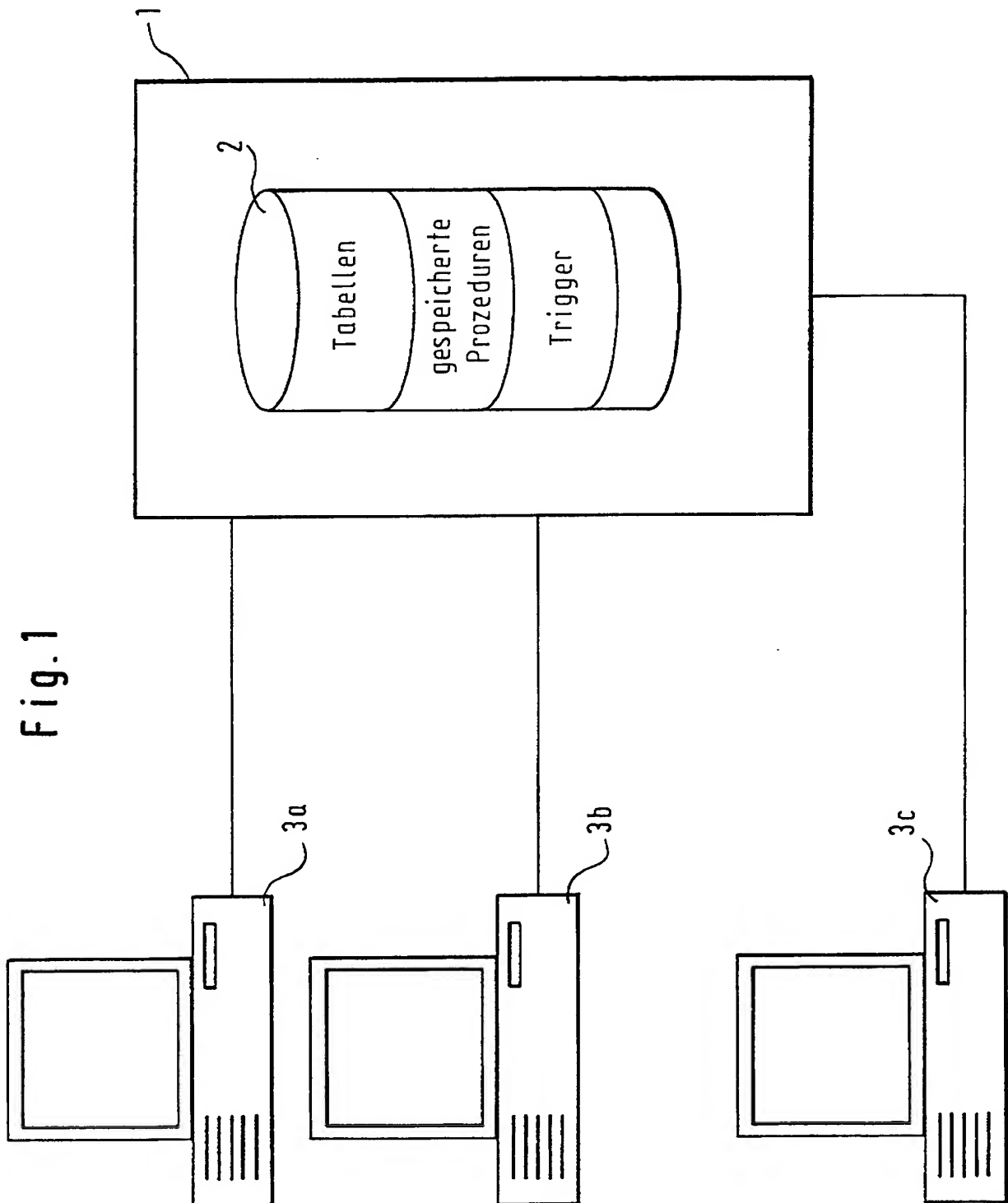


Fig. 2a

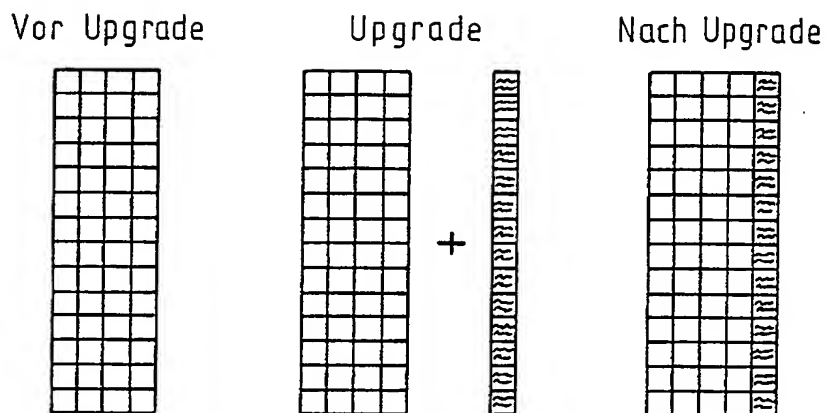


Fig. 2b

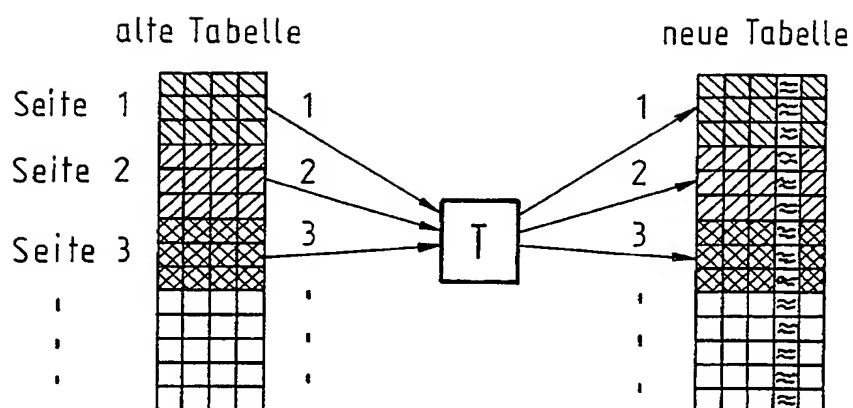


Fig. 3

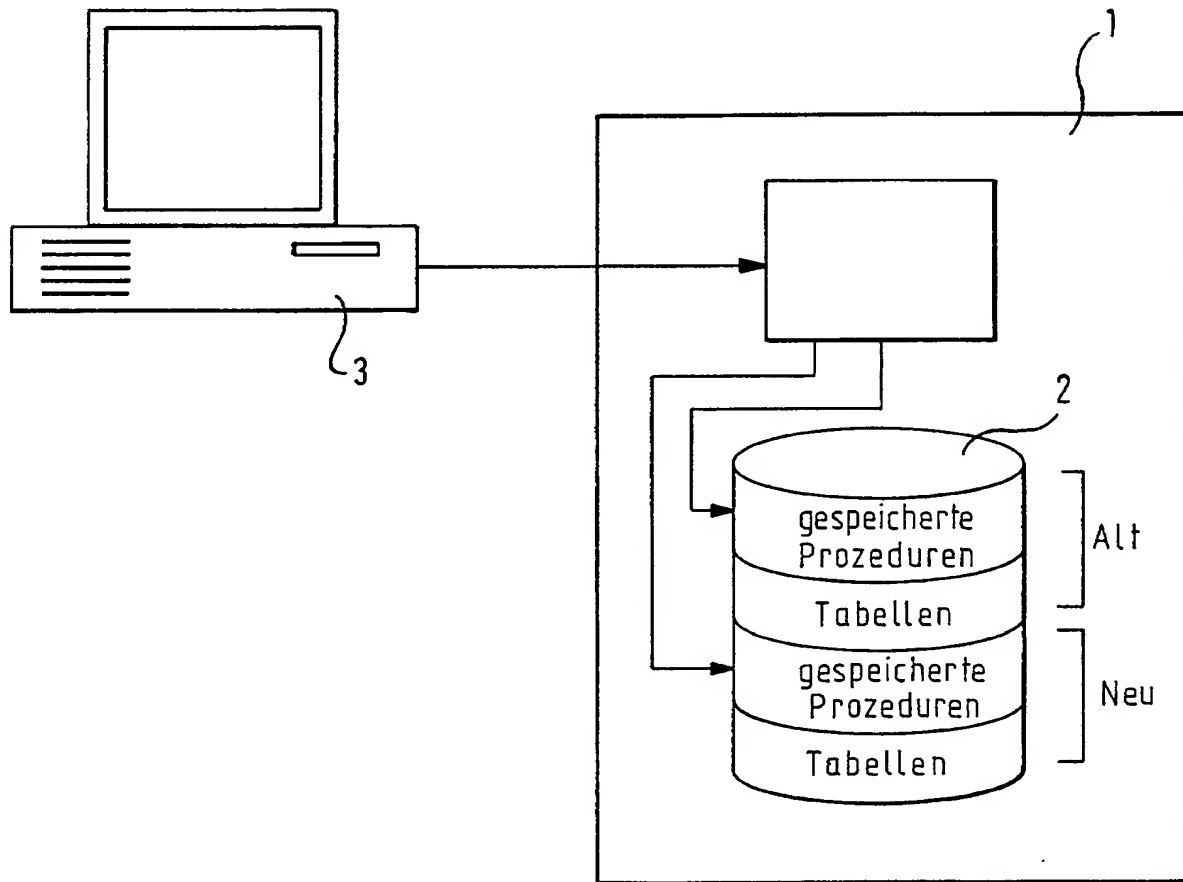


Fig. 4

